APPLICATION

FOR

UNITED STATES LETTERS PATENT

APPLICANT NAME: Deitch

TITLE: SYSTEM AND METHOD FOR MODELING LPAR BEHAVIORS IN A SIMULATION TOOL

DOCKET NO.: CHA920030034US1

INTERNATIONAL BUSINESS MACHINES CORPORATION

CERTIFICATE OF MAILING UNDER 37 CFR 1.10

I hereby certify that, on the date shown below, this correspondence is being deposited with the United States Postal Service in an envelope addressed to Mail Stop Patent Application, Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450 as "Express Mail Post Office to Addressee"
Mailing Label No. EV 393 299 565 US

on February 27, 2004

Beverly Kehoe Shea
Name of person mailing paper

Signature

February 27, 2004
Date

SYSTEM AND METHOD FOR MODELING LPAR BEHAVIORS IN A SIMULATION TOOL

BACKGROUND OF THE INVENTION

1. Technical Field

The present invention relates generally to simulation tools for mainframe computers, and more specifically relates to a system and method for modeling LPAR behaviors within a mainframe simulation tool.

2. Related Art

A logical partition (LPAR) is the division of a computer's processors, memory and storage into multiple sets of resources so that each set of resources can be operated independently with its own operating system instance and applications. The number of logical partitions that can be created depends on the system's processor model and resources available. Typically, partitions are used for different purposes such as database operation or client/server operation or to separate test and production environments. Each partition can communicate with the other partitions as if the other partition is in a separate machine. Both of IBM's® z/OS and AS/400 products support logical partitioning. HITACHI® and SUN MICROSYSTEMS® also use forms of logical partitioning.

Often it is necessary to model the performance of mainframe systems utilizing LPARs, e.g., to meet customer needs, etc. Unfortunately, current modeling techniques do

not account for various LPAR behaviors. One such behavior involves time slice dispatching, which occurs when the total processor consumption nears 100%. Time slice dispatching, as opposed to event dispatching, refers to the way physical CPs (central processors or central processing units) are dispatched to logical CPs by the dispatcher. Because time slice dispatching has an impact on the workload performance of the system, it must be accounted for.

Another behavior involves the interaction of several LPARs running on the same processor. In an LPAR mode configuration, the performance of workloads running in one LPAR can be heavily affected by the work running in other LPARs. This is especially true when the total utilization of the processor approaches 100%. Since known simulators generally only model the workloads in one LPAR at one time, the workloads running in the other LPARs are defined as a constant amount of capacity (i.e., some number of MIPS). However, this assumes the amount of capacity used by these other LPARs is fixed or constant. In real systems, the amount of capacity used by these other LPARs is variable and is a function of the capacity used by the LPAR being modeled.

Accordingly, a need exists for a simulation tool that can better model LPAR behaviors.

SUMMARY OF THE INVENTION

The present invention addresses the above-mentioned problems, as well as others by providing a simulation tool that can model LPAR behaviors associated with operating in time slice dispatch mode and running multiple LPARs. In a first aspect, the invention provides a method for modeling a behavior of an LPAR (logical partition) in a simulated

computer operating in a time slice dispatch mode, comprising: beginning a next modeling interval; calculating a resource percentage representing a percentage of total resources allocated to the LPAR; calculating a time slice percentage for the LPAR based on the resource percentage; determining a CP (central processor) percentage representing a percentage of time that all physical CPs in the computer being modeled have been allocated to the LPAR; and if the CP percentage is greater than the time slice percentage, causing the simulated computer not to dispatch CPs to the LPAR.

In a second aspect, the invention provides a computer simulation tool having a tool for simulating operation of a computer having a system for modeling a behavior of an LPAR operating in a time slice dispatch mode, the system comprising: a system for calculating a resource percentage, wherein the resource percentage represents a percentage of total resources allocated to the LPAR; a system for calculating a time slice percentage for the LPAR based on the resource percentage; a system for determining a CP percentage, wherein the CP percentage represents a percentage of time that all physical CPs in the computer being modeled have been allocated to the LPAR; and a system for determining causing the computer simulation not to dispatch CPs to the LPAR for a current modeling interval if the CP percentage is greater than the time slice percentage.

In a third aspect, the invention provides a program product stored on a recordable medium for modeling a behavior of an LPAR (logical partition) in a simulated computer operating in a time slice dispatch mode, comprising: means for calculating a resource percentage, wherein the resource percentage represents a percentage of total resources allocated to the LPAR; means for calculating a time slice percentage for the LPAR based

on the resource percentage; means for determining a CP percentage, wherein the CP percentage represents a percentage of time that all physical CPs in the computer being modeled have been allocated to the LPAR; and means for determining causing the simulated computer not to dispatch CPs to the LPAR for a current modeling interval if the CP percentage is greater than the time slice percentage.

In a fourth aspect, the invention provides a method for modeling workload performance of a plurality of LPARs (logical partitions) in a computer simulation, comprising: providing a model for each LPAR specified in the computer simulation, wherein each model includes a defined consumption that is dependent on a consumption of the other LPARs; initializing the defined consumption for each model; running each model and determining an observed consumption for each model; comparing the observed consumption with the defined consumption for all of the models; for each model that has an observed consumption that does not agree with the defined consumption, feeding back the observed consumption to the other models; adjusting the defined consumption of each model based on the feedback; and iteratively rerunning each model until the observed consumption agrees with the defined consumption for each model.

In a fifth aspect, the invention provides a computer simulation tool for modeling workload performance of a plurality of LPARs (logical partitions), comprising: a system for building a model for each LPAR specified in the computer simulation, wherein each model includes a defined consumption that is dependent on a consumption of the other LPARs; a system for running each model and determining an observed consumption for each model; a system for comparing the observed consumption with the defined

consumption for all of the models; a system for feeding back the observed consumption to the other models from each model that has an observed consumption that does not agree with the defined consumption; a system for adjusting the defined consumption of each model based on the observed consumption feedback; and a system for iteratively rerunning each model until the observed consumption agrees with the defined consumption for each model.

In a sixth aspect, the invention provides a program product stored on a recordable medium for modeling workload performance of a plurality of LPARs (logical partitions), comprising: means for specifying a model for each of a plurality of LPARs, wherein each model includes a defined consumption that is dependent on a consumption of the other LPARs; means for running each model and determining an observed consumption for each model; means for comparing the observed consumption with the defined consumption for all of the models; means for feeding back the observed consumption to the other models from each model that has an observed consumption that does not agree with the defined consumption; means for adjusting the defined consumption of each model based on the observed consumption feedback; and means for iteratively rerunning each model until the observed consumption agrees with the defined consumption for each model.

In a seventh aspect, the invention provides a computer simulation tool for modeling LPAR behavior, comprising: a first algorithm for modeling the behavior of an LPAR (logical partition) operating in a time slice dispatch mode; and a second algorithm for modeling the behavior of a plurality of LPARs.

BRIEF DESCRIPTION OF THE DRAWINGS

These and other features of this invention will be more readily understood from the following detailed description of the various aspects of the invention taken in conjunction with the accompanying drawings in which:

Figure 1 depicts a simulation system in accordance with an embodiment of the present invention.

Figure 2 depicts a flow diagram of a method for modeling a behavior of an LPAR in a computer simulation operating in a time slice dispatch mode in accordance with an embodiment of the present invention.

Figure 3 depicts a flow diagram of a method for modeling workload performance of a plurality of LPARs in a computer simulation in accordance with an embodiment of the present invention.

DETAILED DESCRIPTION OF THE INVENTION

Referring now to drawings, Figure 1 depicts a simulation system 10 for simulating the operation of an LPAR-based computer system. Simulation system 10 includes two LPAR modeling systems 12 and 14, which model LPAR behavior that commonly occurs in two different situations in order to provide an accurate workload performance 16. As noted above, LPARs (logical partitions) are utilized in various mainframe computers, including IBM® z/OS and AS/400 products, as well as HITACHI® and SUN MICROSYSTEMS® products.

Time slice dispatch modeling system 12 models the behavior of an LPAR when the computer simulation operates in a time slice dispatch mode, which occurs when the

total processor consumption nears 100%. Time slice dispatch modeling system 12 includes various systems that operate at the beginning of each modeling run. These systems include:

1. A first system 20 for calculating a resource percentage, wherein the resource percentage represents a percentage of total resources allocated to the LPAR. The resource percentage (RP%) is calculated as follows,

RP% = 100% - (sum of resources, by percent, that can be used by each of the other LPARs).

The percentage of resources that can be used by the other LPARS may be based on at least three values: (1) a weighting factor, 0-100%, specified for each LPAR; (2) the number of logical CPs allocated to each LPAR; and (3) the MIPs value for each LPAR.

2. A second system 22 for calculating a time slice percentage for the LPAR based on the resource percentage. The time slice percentage provides a percentage of time that a physical CP (central processor) must be allocated to a logical CP (central processor) for an LPAR to guarantee that the physical CP can meet its minimum share of the total processor capabilities. The time slice percentage (TS%) is calculated as follows:

$$TS\% = \frac{RP\% \times (\#\text{ of physical CPs})}{(\#\text{ of logical CPs})}$$

3. A third system 24 for determining a CP percentage (CP%) for the LPAR, wherein the CP percentage represents a percentage of time that all physical CPs in the computer being modeled have been allocated to the LPAR.

4. A fourth system 26 for causing the computer simulation not to dispatch CPs to the LPAR for a current modeling interval if the CP percentage is greater than the time

slice percentage. A more detailed explanation of the overall process is described below with reference to Figure 2.

The second modeling system, LPAR interaction modeling system 14, provides workload performance for the case when a plurality of LPARs is modeled by simulation system 10. Because all LPARs consume resources, all LPARs must be taken into account when modeling the workload performance of each individual LPAR. LPAR interaction modeling system 14 includes the following systems:

1. A first system 28 for building a model for each LPAR specified in the computer simulation, wherein each model includes a defined consumption value that is dependent on a consumption of the other LPARs. The defined consumption value may, for example, be defined by a MIPs parameter associated with each LPAR. The initial consumption value for each LPAR is assumed to be a constant value.

2. A second system 30 for running each model separately and determining an observed consumption for each model.

3. A third system 32 for comparing the observed consumption with the defined consumption for all of the models, (e.g., in MIPs). If the two values fall within a predetermined threshold (e.g., 1.25%), then the two values agree.

4. A fourth system 34 for feeding the observed consumption back to the other models from each model that has an observed consumption that does not agree with the defined consumption.

5. A fifth system 36 for adjusting the defined consumption of each model using the observed consumption feedback. Thus, all other models will correctly reflect the observed value for a given LPAR.

6. A sixth system 38 for iteratively rerunning each model until the observed consumption agrees with the defined consumption for each model. This ensures that the interaction of each LPAR with the other LPARs is correctly modeled. A more detailed explanation of the process is described below with reference to Figure 3.

Referring now to Figure 2, a flow chart is shown depicting a method for modeling a behavior of an LPAR in a computer simulation operating in a time slice dispatch mode. A next modeling interval begins at step S1. A modeling interval may for example occur at ever 1/100 of a second. At step S2, the resource percentage (RP%) is calculated, wherein the resource percentage represents a percentage of total resources allocated to the LPAR. At step S3, the time slice percentage TS% for the LPAR is calculated based on the resource percentage. At step S4, the CP percentage (CP%) is determined, which represents a percentage of time that all physical CPs in the computer being modeled have been allocated to the LPAR. CP percentage can be readily obtained from the simulation system 10 in a manner known in the art.

At step S5, a check is made to determine whether the CP percentage (CP%) is greater than the time slice percentage (TS%). If it is greater, no CPs are dispatched to the LPAR for the next modeling interval. If CP% is not greater than TS%, then CPs are dispatched to the LPAR at step S6 for the next modeling interval.

Referring now to Figure 3, a flow chart is shown depicting a method for modeling workload performance of a plurality of LPARs. At step S10, a model is provided for each LPAR specified in the simulation, wherein each model includes a defined consumption (e.g., MIPs) value that is dependent on the consumption of the other LPARs. At step S11, the defined consumption is set to an initial constant value for each

LPAR. Any method may be used for selecting the initial value. At step S12, each LPAR model is run separately, and an observed consumption is determined for each model. At step S13, a check is made to see if the defined consumption agrees with the observed consumption for each of the models. If the two values are in agreement (i.e., within some predetermined threshold) for each model, then the simulation is complete at step S14.

If one of more of the models have observed/defined consumption values not in agreement, then the observed consumption value for each model not in agreement is fed to all of the other models. Thus, if there are five models M1, M2, M3, M4 and M5, and the observed consumption values for models M1 and M3 do not agree with their respective defined consumption values, then the observed consumption from M1 is fed to M2, M3, M4 and M5, and the observed consumption from M3 is fed to M1, M2, M4 and M5.

Then, at step S16, the defined consumption level for each model is adjusted based on the observed consumption values received from the other models. The method then loops back to step S12, where each LPAR model is rerun, and the process is iteratively repeated until the defined consumption agrees with the observed consumption for all of the models.

It is understood that the systems, functions, mechanisms, methods, and modules described herein can be implemented in hardware, software, or a combination of hardware and software. They may be implemented by any type of computer system or other apparatus adapted for carrying out the methods described herein. A typical combination of hardware and software could be a general-purpose computer system with a computer program that, when loaded and executed, controls the computer system such

that it carries out the methods described herein. Alternatively, a specific use computer, containing specialized hardware for carrying out one or more of the functional tasks of the invention could be utilized. The present invention can also be embedded in a computer program product, which comprises all the features enabling the implementation of the methods and functions described herein, and which - when loaded in a computer system - is able to carry out these methods and functions. Computer program, software program, program, program product, or software, in the present context mean any expression, in any language, code or notation, of a set of instructions intended to cause a system having an information processing capability to perform a particular function either directly or after either or both of the following: (a) conversion to another language, code or notation; and/or (b) reproduction in a different material form.

The foregoing description of the preferred embodiments of the invention has been presented for purposes of illustration and description. They are not intended to be exhaustive or to limit the invention to the precise form disclosed, and obviously many modifications and variations are possible in light of the above teachings. Such modifications and variations that are apparent to a person skilled in the art are intended to be included within the scope of this invention as defined by the accompanying claims.